1. EZMLM/IDX USER'S MANUAL

1.1 Introduction.

This chapter helps the (potential) subscriber to ezmlm/idx lists to learn the basic functions of the mailing list manager. It contains information on how to start and stop subscribing to ezmlm/idx list, how to access the list archive, how to send messages to the list, and how to contact the list owner, if necessary.

Features marked ``(*)'' are available only if the list uses ezmlm-idx in addition to ezmlm. See More information for more information on ezmlm and ezmlm-idx.

ezmlm is a modern mailing list manager. Its purpose is to efficiently send a message to a large number of recipients with minimal delay. It allows automated additions and subtractions from the subscriber database. In addition, it may keep an archive of messages. It can also impose restrictions on what may be sent or retrieved and by whom. Some mailing list managers keep a database of subscriber information and tailor the message specifically for each subscriber. ezmlm sends the same message to all subscriber. This is much more efficient. The benefits to the user are that on average posts to ezmlm lists reach subscribers much faster than they would with other mailing list manager.

The *digest list*(*) is special kind of mailing list extension. Instead of receiving posts, it sends out ``digest'', ordered collections of messages, when a predetermined limit in time or traffic volume has been reached. From the subscriber point of view, the digest list for the mailing list `list@host` is called `list-digest@host`. You use the same commands when subscribing or unsubscribing to a digest as you do when you join of leave the main list. The only difference is the list name you use with your commands (`list-digest@host` and `list@host`, respectively. There is nothing to stop you from being a subscriber of both.

To temporarily leave an ezmlm list, just unsubscribe. Subscribe again later, when you wish to receive messages or digests again. ezmlm processes such requests virtually instantaneously, removing the need for a temporary ``hold'' command.

ezmlm is highly automated. Virtually everything can be done simply, rapidly, and without human intervention. ezmlm encodes your address and the message number in the return address to automatically handle *bounces*, i.e. messages returned to sender because they couldn't be delivered to the subscriber. This can be used to find out the message number, as well as the address you're subscribed under (see Unsubscribing from an ezmlm list).

1.2 Sending commands to an ezmlm/idx mailing list.

ezmlm has a different address for each command. In essence, the address is the list name *and* the command. This is straight forward to interpret and therefore very fast. It also makes it very easy to send commands from the command line without having to worry about the subject and the message body (which are ignored).

Messages to the list address itself are sent to all subscribers and should *never* be used for administrative requests. If you have forgotten everything, send a message to mailinglist-help@example.org for the list mailinglist@example.org. If the instructions returned do not help you solve the problem, send a message to mailinglist-owner@example.org for human help, forwarding a message from the list and including any relevant information.

If you want to interact administratively with the ezmlm mailing list mailinglist@example.org, send an empty message (contents and subject are ignored) to mailinglist-command@example.org, replacing ``command'' with the desired command. For some commands, you can add arguments. Here are a few examples, which will be discussed further below:

mailinglist-subscribe@example.org

> Subscribe me. ezmlm will send you a confirmation request, and once you reply to it a message confirming that you are a subscriber.

mailinglist-subscribe-user=gol.com@example.org

> Subscribe user@gol.com. Here the default subscriber address (the one you send from) is overridden by an argument to the *-subscribe* command. To subscribe any address of type user.name@host.dom just replace the ``@'' in the address with ``='' and add it after a hyphen to the command. In this case mailinglist-subscribe-user.name=host.dom@example.org. Of course, ezmlm will ask user.name@host.dom to confirm the request.

mailinglist-unsubscribe@example.org

> Unsubscribe me. ezmlm will send a confirmation request to your address and remove the address once you reply.

mailinglist-unsubscribe-user=gol.com@example.org

> Unsubscribe user@gol.com.

mailinglist-query@example.org(*)

> Tell me if my address is subscribed to the list or not.

mailinglist-get.123@example.org

> Get message 123 from the archive. If you specify a number that is larger than the highest message number in the archive, you'll receive the latest message.

mailinglist-get@example.org(*)

> Get the latest 30 messages from the archive. If the list has a digest, the messages returned are the messages received since the latest digest (the last message of the latest digest is returned so that there always will be at least one message).

mailinglist-get.45_67@example.org(*)

Get messages 45-67 from the archive. No more than 100 messages can be returned per request. If the first argument is larger than the highest message number in the archive, you'll instead receive the latest 30 or so messages from the archive. For a list with a digest, you'll get the messages that have arrived since the last digest as well as the last up to 30 messages of that digest.

mailinglist-index@example.org(*)

Send subject and author name for the last 100-200 messages. No more than 2000 entries are returned per request.

mailinglist-index.300_399@example.org(*)

Send subject and author of messages 300-399.

mailinglist-thread.45@example.org(*)

Receive an ordered set of all messages that have the same subject as message 45. ``Re:'', etc, do not make a difference, so the returned set is the usually entire thread.

mailinglist-info@example.org(*)

Send me some information about this list (the list owner decides how this command is supported).

mailinglist-faq@example.org(*)

Send me the list of frequently-asked questions (and answers) for this list (the list owner decides how this command is supported).

mailinglist-digest-subscribe@example.org(*)

You guessed it - subscribe to the digest list for mailinglist@example.org, provided there is one. Digest are usually sent about every other day, but this also depends on the message volume and settings adjusted by the list owner.

mailinglist-digest-unsubscribe@example.org(*)

Yes, it's that easy!

1.3 Subscribing to an ezmlm mailing list.

As mentioned previously, ezmlm commands are put into the ADDRESS. To subscribe to mailinglist@example.org, send an (empty) message to mailinglist-subscribe@example.org. ezmlm will send a confirmation request back to you. To confirm your subscription, just reply to the ``Reply-To:'' address. Usually, the ``reply''

function of your mail program will do this. When ezmlm receives your confirmation, it will add your address to the subscriber database and send you a message to tell you that you are a subscriber. If your mail program supports it, you may be able to just click on the address in the message.

The confirmation is used so that only you (and the list administrator) can add/remove your address from the list of subscribers.

Sometimes, you may want to subscribe an address other than the one you are sending from. Let's say you're sitting at john@example.net and that all your mail is forwarded here from harold@example.com (when you are not in the office). You'd like harold@example.com to be the subscriber (so that you get the messages at work). To do this, send mail to: mailinglist-subscribe-harold=example.com@example.org. ezmlm will send the confirmation request to harold@example.com which will be forwarded and reach you at john@example.net. When you reply, harold@example.com is added as a subscriber. The ezmlm verification mechanism tests the only relevant issue: That the person at the subscription address really wants to be a subscriber. After the confirmation reply is received, ezmlm will send a message to the subscription address to let the recipient know that s/he is a subscriber.

Some ezmlm mailing lists may use subscription moderation(*). Subscription to these lists is identical to the process described above, *except* that after confirmation the request is passed on to the list moderator(s). You become a subscriber only after one of the moderators has approved your request. If you want to send any additional information to the moderator, do so in your confirmation message. ezmlm will notify you once you are a subscriber.

1.4 Unsubscribing from an ezmlm mailing list.

You've guessed it already! To unsubscribe from mailinglist@example.org just send a message to mailinglist-unsubscribe@example.org, then reply to the confirmation request. ezmlm will send you a message to let you know that you are no longer a subscriber. ezmlm will also let you know if the address was not in the subscriber list. If so, you are probably subscribed under another address. When this happens, construct an unsubscribe request that contains your subscription address.

If you message contains a ``List-Unsubscribe:'' header, just click on it if your mail reader supports it and send of that message. If you mail reader doesn't support this, copy the entire address from after ``mailto:'' until just before ``>'', paste it into the ``To:'' field of a new message and press ``SEND''. Reply to the confirmation request.

Otherwise, you have to know your subscription address. To find it, look at the very first header of any message from the list. This should contain something like mailinglist-return-2345-harold=example.com@example.org which tells you that the message number is ``2345'' and the subscriber address is harold@example.com. Now you remember that you subscribed a address, and construct the appropriate command address to remove that address from the subscriber database: mailinglist-unsubscribe-harold=example.com@example.org. (You can see that this is a normal unsubscribe request, with a ``-'' followed by your subscription address with the ``@''

replaced by a ``=''. If you like, you can just use the address in the ``Return-Path'' header, and replace the ``return-2345'' with ``unsubscribe''.) Replying to the confirmation request will return the desired acknowledgement that the address has been removed from the subscriber database.

You can also send *-unsubscribe* requests for each potential subscription address and reply to the confirmation requests. The message sent by ezmlm in response to you *-unsubscribe* confirmations will tell you if you *-unsubscribe* attempt was successful.

With ezmlm-idx(*) you can also send mail to mailinglist-query-fred=old.com@example.org, mailinglist-query-gollum=previous.edu@example.org, and so on. For each message, the address will receive a reply telling you whether or not the address is subscribed or not. Construct a ``*-unsubscribe*'' message from the one that gave you a positive reply and you're off the list!

If you for some reason are not successful with these attempts you can as a last resort contact mailinglist-owner@example.org. Please describe your problem and include a FORWARDED message from the list and a list of the possible addresses you might be subscribed under. The list owner will be able to help you get off the list, and can usually figure out your subscription address from the information sent (see Helping users to unsubscribe).

Unsubscribe from subscription moderated lists works the same way as for normal lists. You never need approval to remove your address from a moderated list.

Posting from an alternative address when post are allowed only to subscribers.

When a list is set up to allow posts from subscribers only(*), a post from an address (jonesj@softx.com) may be rejected since this address is not a subscriber (even though mail to the subscriber john@univ.edu reaches you, ezmlm has no way of knowing this). The easiest way to deal with this is to unsubscribe john@univ.edu and subscribe jonesj@softx.com. If this is not possible/desirable, send the addresses in question with a note to mailinglist-owner@example.org. The list owner can add your sender address (in this case jonesj@softx.com) to an extra address lists of non-subscribers allowed to post (and access the archive). The extra addresses are kept in a database much like subscriber addresses. In fact, you can add the address jonesj@softx.com as an alias for the list mailinglist@example.org by mailing mailinglist-allow-subscribe-jonesj=softx.com@example.org and replying to the confirmation request. Again, you're changing the ``target'' address of the request from the default (the sender address) by adding the target to the command with the ``@'' replaced by ``=''. Of course, the ``allow'' list doesn't send out posts. It is solely a vehicle for storing ``allowed'' aliases.

1.5 Sending messages to an ezmlm mailing list.

To send messages to an ezmlm list, you must know its name. If the list is called mailinglist@example.org, just send a message to mailinglist@example.org. Any message accepted at this address is sent to all list subscribers.

Due to the efficiency of ezmlm, you usually get back the message you sent to the list within a few minutes. Of course, this is true only if you are a subscriber.

Most ezmlm lists require that you specify a ``Subject'', a few words that give everyone an idea what your message is about. This is a good thing, since many subscribers automatically delete mailing list messages without subject. Conversely, a good subject causes those interested/able to help with your comment/problem to read the message. If you send a message without a subject, ezmlm will send it back to you with an error message.

ezmlm also rejects messages to the list if the subject is a single command word only, such as ``subscribe'' or ``help''. These are almost always misdirected commands from novice users. Instead of ``flames'' from many subscribers, the sender gets an informative error message. As you know, your should send mail to mailinglist-unsubscribe@example.org to unsubscribe from the list mailinglist@example.org and a message to mailinglist-help@example.org to get help and command information.

The list owner may place additional restrictions on messages to the list. ezmlm allows rejection of messages that are not from subscribers(*), that are too long/short(*), or that contain certain MIME parts(*). If your message is not acceptable by these criteria, ezmlm will send your message back to you and tell you why it failed. You can then correct the problem and try again. You can also mail mailinglist-owner@example.org to reach the list owner and ask for assistance. Replace ``mailinglist'' and ``example.org'' with the real list ``local'' and ``host'' name, respectively. In all messages to the owner, please include a FORWARDED copy of the error/problem message (see below). *Note:* ezmlm lists handle almost everything themselves. Therefore, the list-owner may check the mail only infrequently. Use ezmlm if at all possible, and be patient when you wait for a reply from the owner.

In special cases, the list owner may set up the list to be message moderated(*). When you subscribe, you should be told if the list is moderated. Also, ezmlm adds a special ``Delivered-To: moderator for ...'' header to the messages. On a message moderated list, your message, instead of going directly to the list, is sent to one or more moderators. They can accept or reject, but not modify the message. If the message is accepted, it is sent to the list unmodified. If it is rejected, it is sent back to you, optionally with an explanation from the moderator. On moderated lists it may take a little longer for the message to reach all the subscribers, since it has to be read and approved by at least one moderator before being sent out to the subscribers.

Message moderation is very useful for e.g. announcement lists, or lists where for other reasons not all posts are accepted. ezmlm does not reveal the moderators' identity to you. If you feel that you are treated unfairly by a moderator, please state your case to the list-owner.

1.6 Replying to a message from an ezmlm mailing list.

Messages sent via an ezmlm mailing list come ``From:'' the original sender. For many lists, it is appropriate to discuss privately with the original sender, then post a follow-up message with e.g. a solution to the list. In this case, just ``Reply'' to the sender. Without this behavior two things may happen: First, a subscriber may intend to reply to the original sender only, but sends a message to the entire list. This can be very embarrassing. Second, misconfigured subscriber programs sometimes send ``John is on vacation'' messages back to the ``From:'' address. The ezmlm setup allows these to go to the original sender (a minor nuisance) rather than to the list (a major nuisance with the potential for mail loops if the subscriber software is severely misconfigured).

In other cases, you may want to send the reply to all subscribers. The easiest is to type in the list address by hand or to use the ``Reply-to-all'' function of your mail reader. If you do this, delete your address as well as the original sender's address so that the message goes to the list only. Since messages reach subscribers within minutes, it is usually superfluous and sometimes annoying to also send a reply directly to the sender as well.

1.7 Accessing the ezmlm message archive.

By default, ezmlm mailing lists keep an archive of all the messages sent to the list. Some of them may have been removed by the list owner, but usually at least the last few month's message will be there. With ezmlm you can retrieve one message at a time with the *-get* command. With ezmlm-idx(*) you can access the archive in three different ways: You can get an ``*-index*'' listing message subjects and authors only. This information is sent as sets of 100 messages with up to 2000 entries per request. You can also ``*-get*'' a range of messages(*) (up to 100 per request) which will be sorted by subject (``Re:'' and other ``subject modifiers'' are ignored) and time received. Finally, you can retrieve a set of messages or a ``thread'' containing a specific message. This usually gives you an ordered set starting with the first post, and then all replies in order received.

To access the archive, again remember that commands are put into the ADDRESS. Here are examples of the commands:

mailinglist-get.123@example.org

> Get message 123.

mailinglist-index@example.org(*)

> Get the subject and author information for the last 100-200 messages. This also tells you the latest message number.

mailinglist-index.205@example.org(*)

> Get the index for messages 200-299.

mailinglist-index.200_299@example.org(*)

Get the index for messages 200-299.

mailinglist-get@example.org(*)

Get the messages accumulated since the latest digest (returns latest 30 messages if there are no digests for the list). The normal format is MIME with only the important headers. If you instead of the *-get* command use *-getr* you get the digest as a single long message without MIME. The *-getv* command gives you the MIME format, but now all the headers of the individual messages are returned. (Normally only the important headers are included to save space but other headers may sometimes be relevant for error tracing, etc.)

mailinglist-get.123_456@example.org(*)

Get messages 123-456. Because you can only get up to 100 messages per request, only messages 123-222 will be send. Send new requests to get the remainder.

mailinglist-thread.123@example.org(*)

Get the thread containing message 123. To change the format, use the *-threadr* or *-threadv* format instead. These work just like their *-get* cousins (see above).

If you haven't been following the list for a while, or you just want to see the discussion of a specific question, the easiest is to start with *-index*(*). If not found, send further *-index*(*) requests going backwards in the archive. Once you've found one message (e.g. ``667'') in the thread use the *-thread*(*) command: mailinglist-thread.667@example.org to get the entire set.

If you're a digest subscriber and find an interesting discussion, use *-thread* or the mailinglist-get@example.org address to catch up with the latest messages in that thread or on the list. The *-thread*(*) command is also useful if you catch the tail end of a discussion and want to read earlier messages.

Some lists may be set up to allow archive access to subscribers only(*). If you are not a subscriber, you can simply subscribe. If you are a subscriber, but are denied archive access, you are subscribed under an address different from the one you are sending from. Easiest is to unsubscribe (see above) the current subscriber address and subscribe the address you're sending from. Alternatively, some mail programs (e.g. Mutt) allow you to change the SENDER address depending on where you are sending mail. Last, if this address difference is necessary, you may FORWARD a list message, and a note with the two addresses and a request to mailinglist-owner@example.org. The owner will add your second address to a special address list of senders allowed to access the archive (and post) even though they are not subscribers. Remember to replace ``mailinglist@example.org'' with the list address. (If you are adventurous, see Adding an alias for information on how to do this without the help of the owner.)

1.8 Getting ezmlm digests a few times per week rather than individual messages(*).

ezmlm list may be set up with a digest function. This is basically a second address database used to send out ``digests'', ordered collections of messages sent to the main list. Usually, a ``digest'' is sent out every other day. This is very convenient for users who want to follow a list, but not participate in the discussion. A digest is sorted, so it's easy to read what interests you. Thus, the digest for mailinglist@example.org would be called mailinglist-digest@example.org and you would mail mailinglist-digest-subscribe@example.org.

Simple ezmlm/idx lists test if the conditions for a digest are met whenever a message arrives. Thus, the digests do not arrive at a specific time, but rather approximately every two days, or more often if there is a lot of traffic.

On some mailing lists you are a subscriber, and as a subscriber, you can choose to receive the messages individually or as a digest. With ezmlm, you can subscribe to the ``list'' for individual messages, or to ``list-digest'' for digests. Some people prefer to subscribe to both: They can keep up with a discussion if they like, but if they have too little time they can just delete the individual messages and read the ``threads'' that interest them. To switch from being a ``list'' subscriber to being a ``list-digest'' subscriber, simply subscribe to the digest and unsubscribe from the main list. You will still be able to post to the main list.

With some mailing list managers, the digest is edited by a human. ezmlm digests contain each message as it was sent out by the list (apart from having many less informative headers removed). No editorials, no missing messages. The messages are ordered by subject (or ``story'' or ``thread'') and within each subject by arrival time. First in the digest is a table of contents with subjects, message numbers, and authors. Messages are enclosed as attachments so that a MIME-capable mail reader will allow you to see them as individual messages. This way, you can easily follow-up on a post even from the digest. The owner may chose to use a non-MIME single message format instead. This is most often done on old lists where the subscribers are used to that format and unaware or due to deficient mail readers unable to appreciate the advantages of the MIME format (Pegasus Mail and Mutt are free programs that do a good job).

The digest of a list ``list'' is called ``list-digest'', and the commands are the same as for the main list. Thus, to subscribe to mailinglist-digest@example.org (the digest of mailinglist@example.org) send a message to: mailinglist-digest-subscribe@example.org, or to subscribe the address jonesj@softx.com send to: mailinglist-digest-subscribe-jonesj=soft.com@example.org. Of course, ezmlm will send a confirmation request, and then a message notifying the user that s/he is a subscriber. If the list owner has chosen to restrict posts and archive access to subscribers, digest subscribers are automatically given the same privilege.

1.9 Unsubscribing from an ezmlm digest list.

You unsubscribe to the digest in the same manner as you unsubscribe from the main list (see Unsubscribing from an ezmlm list), but using the digest list name. Thus, to unsubscribe your current address from mailinglist-digest@example.org send mail to: mailinglist-digest-unsubscribe@example.org. Again, you can unsubscribe john@example.com by instead mailing mailinglist-digest-unsubscribe-john=example.com@example.org. ezmlm confirmation works as usual.

1.10 Temporarily halting message delivery.

Just unsubscribe from the list or list-digest, then subscribe when you want mail to resume. If desired, get missed messages from the archive with *-get* or just subject/author info with *-index*(*).

Due to the fast replies from ezmlm this is easier and more efficient than having ``dormant'' subscribers.

1.11 Contacting the list-owner.

You can contact the list owner by mailing to the ``owner'' address. For the list mailinglist@example.org just send your message to mailinglist-owner@example.org. Please state your problem and FORWARD any error message or a list message with your request. Since ezmlm can take care of virtually all request without human intervention, the list owner may read mail only infrequently. Please be patient and remember that in most cases you are asking the list owner to manually do what you could do yourself by e-mail via ezmlm with the information in the error or list message.

2. EZMLM/IDX MODERATOR'S AND ADMINISTRATOR'S MANUAL

2.1 Introduction.

This chapter describes how to be a moderator(*) (message and subscription) and remote administrator(*) for a mailing list run by ezmlm/idx

Features that require ezmlm-idx are marked ``(*)''.
. An ``administrator'' is a person responsible for the management of a mailing list. A ``remote administrator'' is an address with the rights to do certain administrative tasks remotely by E-mail. Depending on how the list is configured, remote administrators may have the right/ability to list subscribers, search for subscriber addresses by name, add and remove subscriber addresses and/or edit the texts that ezmlm sends in reply to list commands. A ``moderator'' is an address to which messages are sent for approval. The message will be passed on only if a moderator has approved it. ezmlm distinguishes between ``message moderators'' and ``subscription moderators''. Posts to a message moderated list are sent to the ``message moderators''. The list acts on the first moderator reply it receives to accept or reject the post. For subscription moderated lists, the users' subscription requests are sent to the ``subscription moderators''. If at least one

subscription moderator approves the request the user address becomes a subscriber. While remote administration and subscription moderation are enabled separately, the addresses are always the same. Message moderators can be a different group of addresses, but in practice most lists use the same addresses for all 3 functions.

In all transactions ezmlm keeps the moderators' addresses secret, even from other moderators. If you want to tell the user who you are, send them a separate E-mail. Of course, a list owner may choose to make the moderators' identities public.

ezmlm also keeps subscriber addresses secret. Only if configured (see below) will it allow the addresses to be listed, and then only to remote administrators.

ezmlm sends special moderator help in response to a *-help* request for a moderator(*). To receive this information for mailinglist@example.org, send mail to mailinglist-help@example.org.

2.2 Message moderation(*).

ezmlm lists may be set up with message moderation. When ezmlm receives a message for the list, it saves it, sends a moderation request with the message enclosed to the moderator(s). The subject of moderation request is ``MODERATE for ...".

If you would like to approve the message, just reply to the ``Reply-To:" address of the moderation request. Usually, this is the default for the ``reply" function of your mail program. Alternatively, click on the ``accept" address or copy and paste it into the ``To:" field of a new message.

To reject the message, reply to the "From:" address, click on the ``reject" address, or copy and paste the ``reject" address into the ``To:" field of a new message. You can also just replace the text ``accept" in the accept address with ``reject". Optionally, type a comment between two lines starting with ``%%%". The sender of the message will receive a message from ezmlm explaining that the message was rejected. If you typed a comment, it will be put into the rejection notice.

The third alternative is to ignore the request. If no moderator has replied within 5 days, the message is returned to the sender with an explanation. The owner can also set up the list so that ignored posts are silently removed, which may be more suitable for some lists.

To minimize mail to busy moderators, ezmlm does not confirm moderator actions. More than one moderator may reply. The message will not be duplicated. The first valid reply that is received will decide the fate of the message. ezmlm will send an error message only if a later reply requests an action different from the one already taken. Thus, if you decide to accept the message, but another moderator has already rejected it, ezmlm will notify you. If instead the message had been previously accepted, ezmlm would log your request, but it will not send you a notification.

If a message is sent *by* a moderator, it is sent for approval to that moderator only. The approval is necessary since anyone can claim to be a moderator, but only the moderator will be able to receive the moderation request. This feature is very useful for

announcement lists where any moderator can make an announcement without bothering the others, while at the same time non-moderators cannot post without permission.

2.3 Remote administration(*).

A remote administrator is an address that can add/remove addresses from the subscriber list. To subscribe john@example.net to the list mailinglist@example.org, send mail to mailinglist-subscribe-john=example.net@example.org. Normally, john@example.net will be asked to confirm. However, if you are a remote administrator, *you* will receive the confirmation request. Thus, john@example.net does not need to participate. When you've successfully added the address, john@example.net will receive a message acknowledging his subscription.

You can unsubscribe addresses (*-unsubscribe*) or determine if an address is a subscriber (*-query*). If the owner has chosen to enable these functions, you may also *-list* subscriber addresses list and/or *-edit* text files, such as the one sent in reply to the *-help* command (see Editing ezmlm administrative messages below). You can also search the subscription log with e.g. the user's name (see Listing and searching the subscription log). You can use all these functions to help users subscribe or unsubscribe to the list. See Helping users unsubscribe in the OWNER'S introduction for details.

2.4 Adding subscriber aliases(*).

ezmlm lists may be set up to only allow subscribers to send messages to the list. This is less secure than moderation, but still keeps most ``garbage'' off the list. Occasionally, a user may wish to send messages from an address other than the subscription address. As a remote administrator, you can add the user's alias to a special ``allow'' database. To add john@example.net as an alias to the mailinglist@example.org, send mail to mailinglist-allow-subscribe-john=example.net@example.org. *-unsubscribe* and other commands work the same way. The messages ezmlm sends talk about the mailinglist-allow@example.org mailing list, but of course you know that this is just a figure of speech.

On lists that do not have subscription moderation, users can add themselves to the ``allow'' database in the same way. This is documented only briefly in the USER'S manual.

Archive access may also be restricted to subscribers. Like subscribers of the list or the digest list, addresses in the ``allow'' database are allowed to access the archive.

2.5 Preventing an address from posting(*).

Just as you may want to allow some non-subscriber addresses to send messages to the list, you may want to (temporarily) prevent an address from posting to the list. This is easily defeated, but very useful if some defective mail robot or other mailing list sends messages to the list address. The system works just like the ``allow'' list, but is called ``deny''. To prevent robot@druga.su from posting to mailinglist@example.org, send mail to mailinglist-deny-subscribe-robot=druga.su@example.org and reply to the confirmation request, respectively.

You can do this only if you are a remote administrator and the list is set up to support this option. You can use *-unsubscribe*, *-query*, and other commands with this address collection, just as for the normal subscriber database. Normal users cannot access the database. This database does not affect archive access.

2.6 Listing subscribers(*).

The list may be configured to allow remote administrators to obtain a list of subscriber addresses. If mailinglist@example.org is set up with this option and you are a remote administrator, mail to mailinglist-list@example.org will get you the list. If you are sending from another account, you can use mailinglist-remote=admin.net@example.org to get the list to remote@admin.net. Of course, this will work only if remote@admin.net is a remote administrator.

Subscriber lists for the ``digest'', ``deny'' and ``allow'' databases are obtained in a similar manner by a message to mailinglist-digest-list@example.org, mailinglist-deny-list@example.org, or mailinglist-allow-list@example.org.

2.7 Subscription moderation(*).

If the list is set up for subscription moderation, a moderator has to approve the user's subscription request. The process starts with a user *-subscribe* request, followed by the normal user confirmation. After this, ezmlm sends a second confirmation request to the moderator(s). If any moderator replies to this request, the user is accepted as a subscriber.

You may send the user a message to request further information. However, this will reveal your address/identity to the user.

Users may unsubscribe at any time without moderator assistance.

2.8 Searching and listing the subscription log(*).

If configured, remote administrators may view or search the subscription log. This is a file that contains all changes in the subscriber database, one per line, in order of occurrence. Each line has the date, the direction (``+'' for additions, ``-'' for removals) and the way the change was made (blank for normal subscribe/unsubscribe, ``manual'' for manual additions from the command line, or ``probe'' for addresses removed via automated bounce handling), the addresses, and for additions usually the ``From:'' header contents from the subscription confirmation reply.

Listing this file allows the remote administrator to see recent additions and removals. This is done by sending a message to list-log@host.

Remote administrators can also search through the log. To retrieve all entries fr_d, where ``_'' can be any character, mail list-log.fr_d@host. The search is case insensitive. When you want to search for addresses, just replace the ``@'' in the address with ``_''. This feature is useful when a user wants to unsubscribe, but has changed his address and doesn't remember the subscription address. Just search for the host name, part of the

user's name, etc. Usually it is obvious which of the few alternatives is the correct one. Construct an unsubscribe request from it, and if you are right, the user will receive a request to confirm the unsubscribe. Replying to it will remove the user from the subscribers.

The command works also for other address databases. Thus, to search through the digest subscriber log for ``Keith'' mail list-digest-log.keith@host.

2.9 Editing ezmlm administrative messages(*).

If configured, remote administrators may edit (via E-mail) ezmlm text files, such as the one sent in reply to the *-help* command. A list of editable files, their use, and instructions for editing may be obtained with the *-edit* command. Thus, for the list mailinglist@example.org, send a message to mailinglist-edit@example.org and follow the instructions. With or without ezmlm-idx, the messages can be modified locally with a text editor.

3. EZMLM/IDX OWNER'S INTRODUCTION

3.1 Introduction.

This chapter tells you how to deal with common problems and introduces you to the power and technical aspects of ezmlm/idx. If you want to customize your list, please use the man pages and FAQ included in the ezmlm and ezmlm-idx packages (see More information).

As the owner of an ezmlm list you will appreciate the high degree of speed and automation of ezmlm. Even with large numbers of subscribers, the parallelism of ezmlm/qmail results in rapid delivery of the message to the subscribers. This makes it easy to have discussions among list members. With slower list managers the discussion is often limited to a small set of users who send mail on the topic to each other, ``Cc:ing'' the message to the list. When the other subscribers receive the message, the discussion is way past this point

Features available only if ezmlm-idx is used are marked ``(*)''.
.

ezmlm *-subscribe* and *-unsubscribe* commands result in a handshake which essentially eliminates the possibility of people other than the user (and possibly remote administrators) adding or removing the user's address from the list.

For the list `mailinglist@example.org`, mail to the `mailinglist-help@example.org` address returns a message with all the relevant ezmlm commands for subscription and message archive related functions. Archive access is simple and powerful.

Together these features make it very easy to set up and manage an ezmlm mailing list. The rare messages to the list-owner address (`mailinglist-owner@example.org`, in the example) are usually due to subscribers who have subscribed under one address, and now use another. Even they can (un)subscribe without owner intervention, but some

subscribers may be unable/unwilling to read the ezmlm help message and follow the instructions therein.

3.2 Using auxiliary address databases(*).

In addition to the regular subscriber address databases for the list and list-digest, ezmlm may use up to three other address databases. One of these the ``allow'' database is housed in **DIR/allow/**

File names are shown in **boldface**. Directories end with a slash. ``**DIR/**'' is the list directory.
. In case posts or access to the archive is restricted to subscribers, addresses present in **DIR/allow/** are also accepted. If the subscriber jonesj@softx.com sometimes posts from john@example.net, you can just add the second address to **DIR/allow/** to permit this. The previous chapter describes how to do this via the ``allow'' command extension (see Adding subscriber aliases). If you have shell access, you can simply:

```
% ezmlm-sub ~/listDIR/allow john@example.net
```

To list all the addresses in **DIR/allow/**:

```
% ezmlm-list ~/listDIR/allow
```

Similarly, you can use ezmlm-unsub(1) and ezmlm-list(1) to remove and list addresses, respectively.

Another address database is **DIR/deny/**. If the list was configured with the ezmlm-make(1) ``-k'' switch, posts from addresses in this database are rejected. This is easy to circumvent, but very handy if a subscriber creates a mail loop (ezmlm/qmail has excellent loop detection, but some subscriber vacation programs generate a new message to the list for every message received and ezmlm has no way of knowing that this is not a legitimate post). For such a subscriber address, you can just temporarily add the sending address to **DIR/deny/**, which is some cases is preferable to simply unsubscribing the user. If this database is configured, remote administrators can access it via the ``deny'' command extension (see Preventing an address from posting).

The final database is **DIR/mod/** in which moderator and remote administrator addresses are stored. Remote administrators may themselves (un)subscribe any address. Remote administration (see Remote administration) is configured with the ezmlm-make(1) ``-r'' switch. The same addresses may also be subscription moderators. If configured (``-s'' switch) a user can subscribe only if one or more of the moderators approve. Messages to the list can also be moderated. This is configured with the ``-m'' switch. Again, the moderators are stored in **DIR/mod/**, although it is possible to configure the list so that message and subscription moderators are different sets of addresses (see More information).

To add a moderator/administrator:

```
ezmlm-sub ~/DIR/mod john@example.net
```

ezmlm-list(1) and ezmlm-unsub(1) are used to list and remove moderators. Normally, the moderator database cannot be accessed remotely. However, remote access can be arranged (see More information).

3.3 How ezmlm list setup is controlled.

ezmlm lists are set up with a one line ezmlm-make(1) command. Existing lists can be modified using the ezmlm-make ``-e'' or ``-+'' (*) switches, without loss of archive, subscriber database, or other state information.

The operation of ezmlm-make is controlled by ezmlmrc(*). To customize ezmlm list setup on a host-wide or user (virtual domain)-wide basis, copy this file from the ezmlm binary directory to **/etc/ezmlmrc** of **~user/.ezmlmrc** and edit it. (See More information.)

3.4 Basic list types.

ezmlm offers virtually infinite possibilities for customization. The more common types of lists can be created and configured with a single ezmlm-make(1) command. For more complicated lists and a discussion of how ezmlm works, see More information).

Public vs. non-public lists.

Normal mailing lists are *public*, i.e. anyone can subscribe/unsubscribe, and access the list archive. Occasionally, you may want to set up a non-public list, where (un)subscribe actions are possible only from the command line or remotely by the administrator(s). This is controlled by the ezmlm-make(1) ``-P'' switch.

Adding a digest list to your ezmlm list(*).

The digest is a collection of messages, produced about every two days. If more than 30 messages or 64 kbytes of ``message body'' have arrived since the latest digest, the digest is sent earlier. All these criteria can be changed (see More information). Also, lists can be set up to produce digests at a specific time each day, every other day, etc (see More information).

Some users do not get involved in discussions on the list but rather want to read messages on a specific topic. For these users, a digest with sorted messages is much preferable to receiving the messages one at a time. These users would subscribe to the digest list, rather than to the list itself.

To set up a digest list with the list, all you need to do is to use the ezmlm-make ``-d'' switch. The digest of the list `list@host` is always called `list-digest@host`. Thus for the list `mailinglist@example.org`, the digest list would be `mailinglist-digest@example.org`.

The digest can be sent in several formats. The default, MIME with removal of ``non-essential'' headers is virtually always the best. A ``format-specifier'' can be used with the ezmlm-get(1) ``-f'' switch to override this. ``v'' returns MIME with all headers, and ``r'' a rfc1153-like non-MIME format (see More information). You can use **DIR/digheaders** to specify the message headers included for archive excerpts when using the ``m'' (default) format.

3.5 The ezmlm archive.

ezmlm mailing lists are by default set up to keep an archive of all the messages sent to the list. This archive is used to create digests. Users can also access messages in the archive, using the *-get*, *-thread*(*), and *-index*(*) addresses (see above). If the users would like to receive messages in a non-default format, they may do so by adding a *format specifier*(*) to the command address name. Thus, to receive messages 45-65 from mailinglist@example.org in the ``v'' (virgin - MIME with all headers) format, the user would send a message to mailinglist-getv45-65@example.org(*).

3.6 Restricting list usage.

ezmlm mailing lists can be set up with many restrictions, both for posts and archive access. Through customization of ezmlm, there are many variations. The more common ones are accessible through ezmlm-make(1) switches when the list is created or modified (see More information).

Placing restrictions on messages to the list.

By default, ezmlm rejects all messages that have an empty subject or a subject consisting of a command name (such as ``subscribe'') only. The lists will also reject messages that do not have the list address in the ``To:'' or ``Cc:'' header(*). This removes the majority of spam without affecting normal posts. In addition, lists may be configured to reject messages below or above a certain size(*), measured on the message body. ezmlm can also reject messages containing certain MIME content types(*). It may make sense to reject messages consisting only of ``text/rtf'' on lists with users on multiple platforms, many of which would be unable to read the message. Similarly, restricting the maximal size of messages to e.g. 20000 bytes may save inexperienced users from quoting entire digests or sending large programs or such as an attachment. All these restrictions are handled by ezmlm-reject(1) in the **DIR/editor** file in the list directory (see More information).

Placing restrictions on who can subscribe(*).

To set up lists with subscription moderation so that each *-subscribe* request must be approved by a moderator, use the ezmlm-make(1) ``-s'' switch.

Lists set up with the ezmlm-make(1) ``-u'' switch allow posts only from subscribers of the list or list-digest. Use of the ``-m'' switch makes all posts moderated (see below). A combination of these switches (``-um'') has a special meaning: posts from subscribers go directly to the list, whereas posts from others are sent out for moderation. Also, messages that are ignored (neither accepted nor rejected) by the moderators, usually result in an informative error message to the sender after about 5 days. However, for ``-um'' lists, such posts are ignored. This gives the moderator the choice of accepting the message, rejecting it with the accompanying notification of the sender, and silently ignoring it (most appropriate for e.g. junkmail/spam).

Like normal subscribers, digest subscribers are allowed to post.

Placing restrictions on archive access(*).

For some lists, you may not want non-subscribers to be able to read archived messages. This restriction can be placed on a list with the ezmlm-make(1) ``-g'' switch. Since ezmlm always sends archive excerpts to the sender of the request this ensures that only persons able to read mail to a subscriber address can receive archive excerpts. All such persons can already read the list traffic. The same is true for digest subscribers who also have access to the digest.

The **DIR/allow/** directory holding the ``allow'' addresses can be useful to give archive access to addresses not in the list or list-digest subscriber databases. Addresses in this database are treated as subscribers, both for archive access and posts, but of course they will not receive list messages.

You can also make the list non-public. This can be done by removing **DIR/public** or by using the ezmlm-make ``-P'' switch (see More information). This results in a list where only remote administrators can access the archive. You can eliminate the archive altogether, but it may be required for other purposes (such as digest generation).

3.7 Modifying messages sent by ezmlm.

All these additions can be made default for all lists by modifying **ezmlmrc** (see More information).

Adding a trailer to each message(*).

You can add small ``trailer'' to each message. This usually would contain information on how to unsubscribe from the list, or where the list *html* archive is kept. To add a trailer, simply put the text in the file **text/trailer** in the list directory. You can use the ezmlm-make(1) ``-t'' switch to set up the list with a simple trailer that gives ``unsubscribe'' instructions. ezmlm will substitute the tags <#h#>, <#l#>, and <#n#> with the list host name, local name, and the current message number, respectively. The current message number for digests is the number of the first message in the digest. You can use this to e.g. add a line ``to get this thread, mail <#l#>-thread<#>@<#h#>''. This will always point to the archive of the current list with the current message number, and thus retrieve the thread containing the current message.

Removing certain headers from messages.

DIR/headerremove contains headers that are removed from all messages before they are sent to the subscribers. Some of these must be removed in order for the list to function. You can add others (see More information). If you add a header that should not occur multiple times, it is good to add the same header here to have it removed from incoming messages, if it exists.

Adding headers to list messages.

Headers in **DIR/headeradd** are added to posts before they are sent to subscribers. They are also added to the archived version of the message.

ezmlm will substitute the tags <#h#>, <#l#>, and <#n#> with the list host name, local name, and the current message number, respectively. For the digest of the list mailinglist@example.org the local name is ``mailinglist-digest''. The current message number for digests is the number of the first message in the digest. You can use this to e.g. add a sequence number header by putting ``X-Seq: <#n#>'' into **DIR/headeradd**.

Stripping certain MIME parts from multipart messages(*).

ezmlm can strip out MIME parts from multipart MIME messages. Some mail clients use ``multipart/alternative'' to send both plain text and html versions of the same message. On many lists this is considered annoying and wasteful (messages are usually 2-3 times as large). You can have ezmlm remove the html part from such messages by adding the line ``text/html'' to **DIR/mimeremove**.

Adding a subject prefix to messages(*).

ezmlm can also add a ``prefix'' to the subject line of each message that does not already contain it. This was common on old lists, when mail readers were unable to sort mail, except by subject. It still popular for historical reasons, even though it violates mail standards. Use this only if you feel that it is necessary, and for novice users where ``look'' is more important than correctness. If you do use this make it the list name and short, e.g. ``[mailinglist]'' (well, that's from the example, in real life this is too long and too long a name for a mailinglist as well. ml@example.org with "ml" would be better). The prefix with the list name can be configured with the ezmlm-make(1) ``-f'' switch. Alternatively, just place the text in the file **DIR/prefix**.

For a real thrill, you can use ``[ml-#]''. Here, ``#'' will be replaced by the message number, labelling all messages with the message number of the first message in the thread. This is not only a mail standard violation, but will inconvenience many users and html archiving programs. Still, it may be useful and help novice users use the archive efficiently (mail ml-thread-#@example.org) where ``#'' is the number in the prefix of the message to get the entire thread from the archive). This works also with partially rfc2047-encoded subjects. (See the ezmlm-send(1) man page for more info.) See Adding a trailer to posts for a better way to point the recipients to the thread retrieval function.

3.8 Allow remote administrators to list subscriber addresses(*).

Many old mailing list managers allow users to list the addresses of all subscribers. This used to be OK, but has lately become a means for junkmailers/spammers to acquire large collections of addresses. Consequently, ezmlm does not allow this. Still, it may be useful for the list owner to be able to get such a list to help a user unsubscribe or to remotely backup the subscriber list. ezmlm will supply such a list if the list was set up with the ezmlm-make(1) ``-r'' and ``-l'' switches *and* the recipient of the list is a ``remote administrator'' (see Remote administration). The command is *-list*. Thus, for the digest of mailinglist@example.org a message to mailinglist-digest-

list@example.org would return the list of subscribers, but only if the sender is a remote administrator and the *-list* function is enabled.

The remote administrator right apply to all address databases of the list (subscribers of list and list-digest, ``allow'' and ``deny''), except the moderator addresses. Moderators' addresses cannot be accessed remotely.

3.9 Allow remote administrators to access the subscription log(*).

The subscription log is stored in **DIR/Log**. An entry is made for each modification made to the subscriber database. For subscription by E-mail, the subscribers ``From:'' line is logged. In contrast to the subscriber database, the log is not written in a crash-proof manner. Thus, information might be lost if the system crashes while the log is being updated.

Remote administrators can list this log with the *-log* command or search it for specific entries with the *-log.xxx* command, where ``xxx'' are letters or numbers to be matched exactly, or an underscore (``_'') to match any character. Since the information is highly overlapping with the subscriber list, it is enabled by the same commands, i.e. the ezmlm-make ``-l'' switch. In addition, it requires remote administration to be enabled (``-r''). See Allowing remote admins to get a subscriber list for more info.

3.10 Allowing remote administrators to edit ezmlm text files(*).

Use the ezmlm-make(1) ``-n'' switch to allow remote editing of text files to remote administrators. Editing is allowed for all *existing* text files. Thus, in order to allow editing of the **DIR/text/info** file (sent in response to the *-info* command), you must at least have created such a file from the shell command line, or via ezmlmrc (default for new lists). See More information for more information.

A list of editable files is contained in the **DIR/text/edit-list** file. This file itself for the list mailinglist@example.org is editable via mailinglist-edit.edit_list@example.org. This file has no effect on which files can be edited, but it useful since it contains the instructions returned in reply to the *-edit* command. Hyphens in file names need to be replaced by underscores in the edit commands. See More information for more information.

3.11 ezmlm handling of undeliverable/bounced messages.

Qmail/mail systems are fault tolerant by virtue of a queuing mechanism where a message is removed from the queue only when the recipient mail transfer agent (MTA) has acknowledged receipt of the message. With this, the responsibility for the message has been transferred from the sending MTA to the receiving MTA. For messages that remain in the queue, the sending MTA will retry the transmission. After a set period of

time, the message will be bounced, i.e. returned as ``undeliverable''. If the recipient does not exist (which may be caused by a misconfiguration of e.g. the domain naming service) the message is returned immediately as ``recipient unknown''.

ezmlm uses the subscriber address and message number encoded within the sender address to keep track of which message have ``bounced'' for which recipient. Since messages can bounce for a variety of reasons, ezmlm doesn't remove a subscriber from the list after a single bounce. Rather, it waits about 12 days after the first bounce, then sends a ``warning'' message to the subscriber. The warning message lists the numbers of the messages missed. The user can retrieve these from the archive. If this warning message bounces as well, ezmlm waits another 12 days, then sends a ``probe''. First when this message bounces is the subscriber removed from the subscriber database. This mechanism guarantees that a subscriber won't be removed due to a temporary error, while assuring that all non-existent or permanently failing addresses will be removed eventually.

3.12 Helping users unsubscribe.

Some users never learn and don't want to spend the time to figure out how to unsubscribe. As a remote administrator(*), you can unsubscribe them entirely on your own (see Remote administration). You can also do this from the command line. Alternatively, you can as a normal user send a *-unsubscribe* request for the subscriber address and let the subscriber confirm. Another alternative is to tell the user to send a message to `listname-help@listhost` and ask the user to follow the instructions, which are very explicit.

Other users have attempted to unsubscribe, but failed. They usually will tell you what they've tried, and forward a list message to you. Most likely, they are subscribed with an address other than their normal sender address. If you're lucky, the subscriber address is in the users E-mail to you. If not, you can ask the user to give you a list of possible addresses. You can then send mail to `listname-list@listhost`(*) (if enabled) and use ``find'' or ``grep'' to search through the list for parts of the users addresses. If remote subscriber listing is not available, try:

```
% ezmlm-list DIR | grep -i 'part_of_user_address'
```

Then send an unsubscribe for the address(es). The user will usually receive a confirmation request (if you found the correct address), and can reply to it to get off the list.

Another possibility is to search the subscription log for a part of the subscribers name, usually shown in the ``From:'' header (see Searching and listing the subscription log).

3.13 Helping subscribers post and access the archive on subscriber-only lists(*).

Subscriber-only lists allow posts and archive access only to subscribers of the main list or the digest list. If a subscriber's posts or archive access attempts are rejected, the subscriber is working with a sender address that is not the one s/he is subscribed under. Usually, the user's sender address will be in the E-mail to you. ezmlm-idx keeps a

database of ``allow'' addresses which are treated like subscribers for the purpose of subscriber-only restrictions, but *are not* sent posts. To add an address, send mail to listname-allow-subscribe-user=userhost@listhost. If you're a remote administrator the confirmation request will come to you, otherwise to user@userhost. Replying to the request will add the address. The user should now be able to post and access the archive. To remove an address, user the same procedure, but with the *-unsubscribe* command. Users can manipulate the database themselves, but this may be too complicated for all but advanced users.

If you set up the list with ezmlm-make ``-umr'' messages from addresses that are not subscribers will be sent for moderation. This allows you to ignore junk mail and approve other posts, as well as add the address to the ``allow'' address database, all without the subscriber noticing (see Placing restrictions on subscribers for more info).

3.14 Stopping mail loops.

Qmail and ezmlm have built-in advanced mail loop prevention. However, some auto-responders are misconfigured and create a new message to the list for every message received. Since this is a completely new message, ezmlm has no way of knowing that it is not a post. This is very rare, but may happen. The symptom is that a ``John is on vacation'' messages will arrive at the list, go out to subscribers, only to be followed by another ``John is on vacation'' message, and so on.

To stop this, look at the sender of the ``vacation'' message. Unsubscribe the address.

Even rarer is that there is a mail loop somewhere on the Internet, and that it sends of messages to your list. Unsubscribing doesn't help. Your only recourse is to block the sender of the message from posting. Easiest is to use the ``deny''(*) function with the ezmlm-make(1) ``-k'' switch, then add the sender address:

```
% ezmlm-sub DIR/blacklist badname@badhost
```

or if configured, send mail as a remote administrator to listname-deny-subscribe-badname=badhost@listhost. You could also add a line:

```
  |/bin/echo -- "$SENDER" |/usr/bin/grep -iv '^badname@badhost$' ||
exit 99
```

to **DIR/editor** of the list.

3.15 Dealing with SPAM.

ezmlm/idx lists by default(*) reject messages that do not have the list address in the ``To:'' or ``Cc:'' headers. Most ``bulk E-mail'' does not pass this test. If you still receive too much junk mail to your list, you can set up your list as a subscriber-only(*) list. This creates a small overhead of users who try to post from addresses that are not their subscriber address. See above on how to deal with this.

Since the message sender is easy to fake, this is not secure. A determined SPAMMER can subscribe to the list or just send mail ``From:'' a subscriber address. The next level of countermeasure is to have a message moderated(*) (and possibly subscription moderated(*)) list. The overhead is higher - someone has to approve each message - but it is much more secure. ezmlm-idx has support for multiple moderators making it easy

to set up a list with fast and efficient moderation, which is a very reliable way to keep junk mail off a list.

3.16 Dealing with archiving servers.

Several sites on the Internet subscribe to mailing lists, archive the messages, and make them publicly available. This may not be desirable, as the this often exposes the addresses of the subscribers. The subscribers, while happy to discuss their work on a peer list, may not appreciate messages from the general public asking for follow-ups to messages posted to the list.

ezmlm lists, by default(*), are set up with a ``X-No-Archive: yes'' header. For older lists or lists using ezmlm alone, just add this header to **DIR/headeradd**. Well-behaved archiving servers respect this. If you want to maintain full control, the only recourse is to make the list subscription moderated(*). Of course, the messages are no more ``secure'' than the least reliable subscriber. Remove the ``X-No-Archive: yes'' header for **DIR/headeradd** if you want to allow well-behaved archivers to collect your messages.

3.17 Making an existing list message or subscription moderated(*).

Like other list changes, you can use ezmlm-make with the ``-e'' edit switch together with other desired switches and arguments. An existing list can be made message moderated with:

```
% ezmlm-make -em DIR
```

To add remote administration, add the ``-r'' switch. To add subscription moderation, add the ``-s'' switch.

3.18 Backing up and removing parts of the archive.

The archive is a set of files, one per message, kept in the **archive** subdirectory of the list directory. The directory **archive/0/** contains messages 1-99, **archive/1/** messages 100-199, etc. The files within the directories are named by the last 2 digits of the message number, ``01-99'' for **archive/0/** and ``00-99'' for others. In addition, each set of messages is indexed(*) in the file **index** stored with the corresponding message files.

With time, the archive grows and it may become necessary to remove some messages. When the list is at message ``12301'' one may wish to remove messages 1-9999 from the archive. Easiest is to remove messages as entire subdirectories. For the above, do:

```
% cd ~/
% tar -cvf listarch1_9999.tar DIR/archive/? DIR/archive/1?
% gzip listarch1_9999.tar
% rm -rf ~/DIR/archive/? ~/DIR/archive/1?
```

The ``dir'' here is the list directory. (You may be able to combine the ``gzip'' action with the ``tar'' command by adding the ``z'' switch.)

Save **listarch1_9999.tar.gz** in a safe place. (Usually, you would choose a better name for the backup ``tar'' file). Nothing else needs to be done. If instead you remove individual messages and use ezmlm-idx, it is useful, but not necessary to do:

```
% ezmlm-idx ~/dir
```

to re-index the archive. (For a large archive on a slow machine, this command may take a few minutes.)

3.19 Restoring parts of the archive.

To restore the messages saved in the example above to the archive, simply:

```
% cd ~/
% gunzip listarch1_9999.tar.gz
% tar -xvf listarcg1_9999.tar.gz
% ezmlm-idx ~/dir
```

Re-indexing the archive (as shown) is recommended, since the format for the **index** file may have been changed if ezmlm was upgraded since the backup.

4. More information.

4.1 More about ezmlm and ezmlm-idx.

ezmlm is a mailing list manager written by Dan J. Bernstein. ezmlm-idx is an extension to ezmlm written by Frederik P. Lindberg. ezmlm has the essential features of a mailing list manager. ezmlm-idx adds useful, but non-essential features like digests, list moderation, etc. The list owner will have made the trade off of features vs. increased size and complexity. To test if your list has ezmlm-idx installed, send a message to `list-get@host` where ``list@host'' is the name of the list. A list with ezmlm alone will send you a help message saying that your message wasn't sent to one of its command addresses. A list with ezmlm-idx will return one or more messages from the archive, or a message telling you that the list is not archived. Throughout this manual, features available only with ezmlm-idx are marked with ``(*)''.

For more technical information about ezmlm, it's options, and error resolution, please read the FAQ (http://www.ezmlm.org/) and the ezmlm man pages. A text version of the FAQ is included in the ezmlm-idx distribution. The latest FAQ in various formats can be found at ftp://ftp.id.wustl.edu/pub/patches/ or ftp://ftp.ezmlm.org/pub/patches/. The FAQ also contains information on where to obtain ezmlm, where to find mailing lists about ezmlm, etc. While it is called a ``FAQ'', the document will with time become the ezmlm/idx technical manual. The man pages are always installed with ezmlm. To read the man page for ezmlm, do:

```
% man ezmlm
```

To read the man page for ezmlm-make, do:

```
% man ezmlm-make
```

Also, take a look at Dan J. Bernstein's pages on ezmlm and other programs he has written at http://www.pobox.com/~djb/ and the qmail home page at http://www.qmail.org/.

4.2 More about this document.

This document is updated for ezmlm-0.53 and ezmlm-idx-0.32. It is available at ftp://ftp.id.wustl.edu/pub/patches/ezman/ as ``ezman-x.xx.format[.tar].gz''. ``x.xx'' is the version, ``format'' is the format, e.g. ``ps'', and ``.tar'' is present if the document consists of several files. ezmlm-idx versions have a 2 digit minor version number, with a third digit added for bug fixes (no new features). Thus, 0.314 is the forth bug fix release of 0.31. Manuals and FAQ follow a similar convention with the first two digits matching the version of the package. Thus, you should always get the documentation matching the first two digits (``xx'' in ``0.xxy''), and there chose the highest ``y'' available.

This document comes with absolutely no warranties. If there are errors or omissions in this manual, please send corrections/comments to the author. The manual is intended to be complete for users and moderators/administrators. For list owners, it is intended more as a summary and introduction to the FAQ.

Copyright and translations.

I reserve the right to sell a printed version of this document in the original English, or in any other language. You may distribute it in any other way, even for money, and of course you may give away a printed version.

If you make modifications, you must retain the original copyright notice and identify clearly the sections you've modified. I'd prefer if you notified me of errors/omissions so that I can improve this document myself.

If you translate this document, you own the translation. You may distribute your translation in any way, including printed and for sale. However, I reserve the right to make and distribute an independent translation.

These restrictions are intended to encourage translation, while making it possible to use this text in e.g. a printed book for sale, and publish such a book in any language. The remainder of this notice is intended to make it as easy as possible to disseminate this document in the wonderful world of free software.

© Copyright 1997, 1998, 1999 Fred Lindberg, lindberg@id.wustl.edu & Fred B. Ringel, fredr@rivertown.net
This page was last built on 10/3/99; 1:26:47 PM on the MacOs
Comments/Suggestions: webmaster@ezmlm.org